



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

A

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/708,722	11/09/2000	Stephan J. Jourdan	2207/9800	2194
25693	7590	12/02/2005	EXAMINER	
KENYON & KENYON (SAN JOSE) 333 WEST SAN CARLOS ST. SUITE 600 SAN JOSE, CA 95110			LI, AIMEE J	
			ART UNIT	PAPER NUMBER
			2183	
DATE MAILED: 12/02/2005				

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)
	09/708,722	JOURDAN ET AL.
	Examiner	Art Unit
	Aimee J. Li	2183

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --
Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1) Responsive to communication(s) filed on 07 September 2005.
 2a) This action is FINAL. 2b) This action is non-final.
 3) Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4) Claim(s) 1-19 is/are pending in the application.
 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
 5) Claim(s) _____ is/are allowed.
 6) Claim(s) 1-19 is/are rejected.
 7) Claim(s) _____ is/are objected to.
 8) Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9) The specification is objected to by the Examiner.
 10) The drawing(s) filed on _____ is/are: a) accepted or b) objected to by the Examiner.
 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
 11) The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12) Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
 a) All b) Some * c) None of:
 1. Certified copies of the priority documents have been received.
 2. Certified copies of the priority documents have been received in Application No. _____.
 3. Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)	4) <input type="checkbox"/> Interview Summary (PTO-413)
2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)	Paper No(s)/Mail Date. _____
3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____	5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152)
	6) <input type="checkbox"/> Other: _____

DETAILED ACTION

1. Claims 1-19 have been examined.
2. In view of the Appeal Brief filed on 07 September 2005, PROSECUTION IS HEREBY REOPENED. The rejections are set forth below.
3. To avoid abandonment of the application, appellant must exercise one of the following two options:
 - (1) file a reply under 37 CFR 1.111 (if this Office action is non-final) or a reply under 37 CFR 1.113 (if this Office action is final); or,
 - (2) request reinstatement of the appeal.
4. If reinstatement of the appeal is requested, such request must be accompanied by a supplemental appeal brief, but no new amendments, affidavits (37 CFR 1.130, 1.131 or 1.132) or other evidence are permitted. See 37 CFR 1.193(b)(2).

Papers Submitted

5. It is hereby acknowledged that the following papers have been received and placed on record in the file: Appeal Brief as filed 07 September 2005.

Claim Rejections - 35 USC § 101

6. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.
7. Claim 1 is rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. Claim 1 is directed towards an “instruction segment” with an intended use of being stored in an instruction cache. “Instruction segment” is defined within Applicants’ specification on page 1, lines 16-17 as “...sequences of dynamically executed instructions that

are assembled into logical units.” This makes the claim non-statutory, since the claim is essentially for instructions, e.g. a program. The intended use of storing the instructions in a cache does not make the claim statutory. Also, the claim language makes it unclear whether the claim is directed to an abstract idea or a practical application.

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

9. Claims 1, 3-8, 12-15 and 17-19 are rejected under 35 U.S.C. 103(a) as being unpatentable over Patel et al., *Improving Trace Cache Effectiveness with Branch Promotion and Trace Packing*, in further view of Johnson, U.S. Patent No. 5,924,092.

10. Regarding claim 1, Patel has taught an instruction segment comprising a plurality of instructions stored in sequential positions of a cache line (see Col.1 line 26 – Col.2 line 15). Patel has not explicitly taught storing the plurality of instructions in sequential positions of a cache line in reverse program order.

11. However, Johnson has taught the storing of blocks of data in reverse order so that those blocks that were in the first block of a data structure that are frequently accessed and modified will require less moving and fewer modifications after being placed at the last location of data structure, the fewer modifications resulting in improved performance (see Johnson, Col.4 lines 13-24). Because the traces of Patel are indexed, as well as accessed and modified, via their head (first) instructions (see Patel, Col.4 lines 23-31 and Col.5 lines 12-18), one of ordinary skill in

the art would have found it obvious to modify the instruction segment of Patel to store the instructions of the instruction trace in reverse order so that the frequently accessed and modified head of the trace will be moved and modified fewer times so that performance is improved.

12. Regarding claim 3, Patel in view of Johnson has taught the instruction segment of claim 1, wherein the instruction segment is a trace (see Patel, Col.3 lines 2-12).

13. Regarding claim 4, Patel in view of Johnson has taught the instruction segment of claim 1, wherein the instruction segment is a basic block (see Patel, Col.2 lines 2-5).

14. Regarding claim 5, Patel has taught a segment cache (see “trace cache” of Fig.1) for a front-end system in a processor, comprising a plurality of cache entries to store instruction segments (see Col.1 line 26 – Col.2 line 15). Patel has not explicitly taught storing the instruction segments in reverse program order.

15. However, Johnson has taught the storing of blocks of data in reverse order so that those blocks that were in the first block of a data structure that are frequently accessed and modified will require less moving and fewer modifications after being placed at the last location of data structure, the fewer modifications resulting in improved performance (see Johnson, Col.4 lines 13-24). Because the traces of Patel are indexed, as well as accessed and modified, via their head (first) instructions (see Patel, Col.4 lines 23-31 and Col.5 lines 12-18), one of ordinary skill in the art would have found it obvious to modify the instruction segment of Patel to store the instructions of the instruction trace in reverse order so that the frequently accessed and modified head of the trace will be moved and modified fewer times so that performance is improved.

16. Regarding claim 6, Patel in view of Johnson has taught an apparatus comprising:

a. An instruction cache system (see Patel, “instruction cache” of Fig.1),

- b. An instruction segment system, comprising:
 - i. A fill unit (see Patel, "fill unit" of Fig.1) provided in communication with the instruction cache system, the segment cache of claim 5 included therein (see Patel, Fig.1),
- c. A selector (see Patel, "selection logic" of Fig.1) coupled to an output of the instruction cache system and to an output of the segment cache (see Patel, Fig.1).

17. Regarding claim 7, Patel in view of Johnson has taught the front-end system of claim 6, wherein the instruction segment system further comprises a segment predictor (see Patel, "multiple branch predictor" of Fig.1) provided in communication with the segment cache. Here, when the multiple branch predictor is coupled with the trace cache and mediated by the selection logic, it effectively predicts segments because the basic blocks stored in the trace cache all begin with branches (see Patel, Col.5 lines 5-29).

18. Regarding claim 8, Patel has taught a method for storing instruction segments in a processor, comprising:

- a. Building an instruction segment based on program flow (see Col.1 line 26 – Col.2 line 15),
- b. Storing the instruction segment in a cache (see Col.1 line 26 – Col.2 line 15).

19. Patel has not explicitly taught wherein the instruction segment is stored in reverse program order.

20. However, Johnson has taught the storing of blocks of data in reverse order so that those blocks that were in the first block of a data structure that are frequently accessed and modified will require less moving and fewer modifications after being placed at the last location of data

structure, the fewer modifications resulting in improved performance (see Johnson, Col.4 lines 13-24). Because the traces of Patel are indexed, as well as accessed and modified, via their head (first) instructions (see Patel, Col.4 lines 23-31 and Col.5 lines 12-18), one of ordinary skill in the art would have found it obvious to modify the instruction segment of Patel to store the instructions of the instruction trace in reverse order so that the frequently accessed and modified head of the trace will be moved and modified fewer times so that performance is improved.

21. Regarding claim 12, Patel in view of Johnson has taught the method of claim 8, wherein the instruction segment is a trace (see Patel, Col.3 lines 2-12).

22. Regarding claim 13, Patel in view of Johnson has taught the method of claim 8, wherein the instruction segment is a basic block (see Patel, Col.2 lines 2-5).

23. Regarding claim 14, Patel has taught a processing engine, comprising:

- a. A front-end stage to build and store instruction segments (see Col.1 line 26 – Col.2 line 15),
- b. An execution unit (see “HPS Execution Core” in Fig.1) in communication with the front end stage (see Col.5 line 30 – Col.6 line 4).

24. Patel has not explicitly taught building and storing the instruction segments in reverse program order.

25. However, Johnson has taught the storing of blocks of data in reverse order so that those blocks that were in the first block of a data structure that are frequently accessed and modified will require less moving and fewer modifications after being placed at the last location of data structure, the fewer modifications resulting in improved performance (see Johnson, Col.4 lines 13-24). Because the traces of Patel are indexed, as well as accessed and modified, via their head

(first) instructions (see Patel, Col.4 lines 23-31 and Col.5 lines 12-18), one of ordinary skill in the art would have found it obvious to modify the instruction segment of Patel to store the instructions of the instruction trace in reverse order so that the frequently accessed and modified head of the trace will be moved and modified fewer times so that performance is improved.

26. Regarding claim 15, Patel in view of Johnson has taught the processing engine of claim 14, wherein the front-end stage comprises:

- a. An instruction cache system (see Patel, "instruction cache" of Fig.1),
- b. An instruction segment system, comprising:
 - i. A fill unit (see Patel, "fill unit" of Fig.1) provided in communication with the instruction cache system (see Patel, Fig.1),
 - ii. A segment cache (see "trace cache" of Fig.1),
- c. A selector (see Patel, "selection logic" of Fig.1) coupled to an output of the instruction cache system and to an output of the segment cache (see Patel, Fig.1).

27. Regarding claim 17, Patel in view of Johnson has taught the method of claim 15, wherein the instruction segments are traces (see Patel, Col.3 lines 2-12).

28. Regarding claim 18, Patel in view of Johnson has taught the method of claim 15, wherein the instruction segments are basic blocks (see Patel, Col.2 lines 2-5).

29. Regarding claim 19, Patel in view of Johnson has taught the method of claim 15, wherein the instruction segment cache system further comprises a segment predictor (see Patel, "multiple branch predictor of Fig.1) provided in communication with the segment cache. Here, when the multiple branch predictor is coupled with the trace cache and mediated by the selection logic, it

effectively predicts segments because the basic blocks stored in the trace cache all begin with branches (see Patel, Col.5 lines 5-29).

30. Claims 2, 9-11 and 16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Patel et al., *Improving Trace Cache Effectiveness with Branch Promotion and Trace Packing*, in view of Johnson, U.S. Patent No. 5,924,092, in further view of Peled et al., U.S. Patent No. 6,076,144.

31. Regarding claim 2, Patel in view of Johnson has taught the instruction segment of claim 1, but has not explicitly taught wherein the instruction segment is an extended block.

32. However, Peled has taught trace segments which have multiple entry points and a single exit that allow redundant code segments to be eliminated from the trace cache, thereby improving cache utilization (see Peled, Col.1 lines 60-63, Col.4 lines 13-37, and Fig.3). Because the specification has defined an extended block to have multiple entry points and a single exit point (see p.2 of Specification), one of ordinary skill in the art would have found it obvious to modify the instruction segments of Patel to allow for multiple entry points and a single exit so that redundant code segments could be eliminated from the trace cache and performance could be improved.

33. Regarding claim 9, Patel in view of Johnson have taught the method of claim 9, but have not explicitly taught wherein the method further comprises:

- a. Building a second instruction segment based on program flow,
- b. If the first and second instruction segments overlap, extending the first instruction segment to include non-overlapping instructions from the second instruction segment.

34. However, Peled has taught building a second instruction segment based on program flow and subsequently extending the first instruction segment to include the non-overlapping instructions from the second instruction segment if the two segments overlap (see Col.4 lines 13-37) in order to reduce the degree of code redundancy in the trace cache (see Col.1 lines 60-63). One of ordinary skill in the art would have recognized that it is desirable to reduce redundancy within a trace cache so that the cache can be more effectively used and more different traces stored. Therefore, one of ordinary skill in the art would have found it obvious to extend an existing instruction segment to include non-overlapping instructions from a second instruction segment in order to reduce trace cache redundancy.

35. Regarding claim 10, Patel in view of Johnson in further view of Peled has taught the method of claim 9, but has not explicitly taught wherein the extending comprises storing the non-overlapping instructions in the cache in reverse program order in successive cache positions adjacent to the instructions from the first instruction segment.

36. However, Patel in view of Johnson has taught that instructions in instruction segments are stored in reverse program order (see paragraphs 21-23 above). Because, an extended segment is still an instruction segment, one of ordinary skill in the art would have found it obvious to also store the extended instruction segments in reverse program order.

37. Regarding claim 11, Patel in view of Johnson has taught the method of claim 8, but has not explicitly taught wherein the instruction segment is an extended block.

38. However, Peled has taught trace segments which have multiple entry points and a single exit that allow redundant code segments to be eliminated from the trace cache, thereby improving cache utilization (see Peled, Col.1 lines 60-63, Col.4 lines 13-37, and Fig.3). Because

the specification has defined an extended block to have multiple entry points and a single exit point (see p.2 of Specification), one of ordinary skill in the art would have found it obvious to modify the instruction segments of Patel to allow for multiple entry points and a single exit so that redundant code segments could be eliminated from the trace cache and performance could be improved.

39. Regarding claim 16, Patel in view of Johnson has taught the method of claim 15, but has not explicitly taught wherein the instruction segments are extended blocks.

40. However, Peled has taught trace segments which have multiple entry points and a single exit that allow redundant code segments to be eliminated from the trace cache, thereby improving cache utilization (see Peled, Col.1 lines 60-63, Col.4 lines 13-37, and Fig.3). Because the specification has defined an extended block to have multiple entry points and a single exit point (see p.2 of Specification), one of ordinary skill in the art would have found it obvious to modify the instruction segments of Patel to allow for multiple entry points and a single exit so that redundant code segments could be eliminated from the trace cache and performance could be improved.

Response to Arguments

41. Examiner would like to note that, while the Examiner is maintaining the same art rejection, prosecution after filing of Appeal Brief was re-opened due to concerns under 35 U.S.C. 101 Non-Statutory Subject Matter, which have not been made clear upon the record. This is to ensure that all possible issues are broached and responded to prior to ruling by the Board of Appeals and Interferences.

42. Applicant's arguments filed 07 September 2005 have been fully considered but they are not persuasive. Applicant argues in essence on pages 6-10

Applicants submit it is unclear how accessing the first instruction in a trace more frequently than the second instruction, and accessing the second instruction more frequently than the third, and so on, would improve the performance of a trace... Since the trace defines sequential instructions, which perform a particular operation, accessing the instructions in decreasing frequency would in no way advance the completion of the particular operation. Indeed, such access of the trace would hinder the completion, thereby defeating the purpose of the trace.

43. This has not been found persuasive. Johnson has taught that the static sorting algorithm stores elements in reverse order, since it stands to reason that the elements first added to the array would be accessed and used before the elements most recently added to the array. By grouping these types of instructions together, the fetch rate of Patel is improved. Patel has taught that not only branch promotion but also trace packing. Trace packing allows an entire segment to be filled when fetched even if it means that the segment takes instructions from two different blocks, i.e. split blocks. Applicant is correct that, in a trace cache, the instructions are executed sequentially, but there is no guarantee that the instruction segment is completely filled. Patel has taught that the instruction segment can be completely filled by, essentially, taking instructions from the next segment and inserting them into the empty slots of the current segment. The type of action requires that the instruction segments, in turn, update whether all instruction slots within the next segment are occupied and where the next segment exactly begins. This means that the segments need to be updated, i.e. modified, with the most current instruction control

information. Johnson has taught that entries that were placed at the beginning of an array, e.g. placed first within the array, are more likely to be accessed first, so, by placing these elements at the end of an array where there are less elements dependent on that one particular element to be made when the elements are modified, such as when Patel's instruction segments are fetched in split groups, less time is required to update the element. Also, with regards to Johnson's teaching about moving elements, when empty slots appear, eliminating them is easier with there are not as many elements to move forward within the data array. This is similar to Patel's trace cache. When empty instruction segments and slots appear in the cache it takes less time to move fewer elements to eliminate the holes in the instruction slots, e.g. it takes less time to move fewer elements forward among split blocks. If more elements need to be moved forward, such as when the segments are filled in the beginning of the cache, then more time is required to realign the segments.

Conclusion

44. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. Applicant is reminded that in amending in response to a rejection of claims, the patentable novelty must be clearly shown in view of the state of the art disclosed by the references cited and the objections made. Applicant must also show how the amendments avoid such references and objections. See 37 CFR § 1.111(c).

- a. Kyker et al., U.S. Patent Number 6,594,734, has taught a trace cache and its operation and how the program order is maintained with pointers to next segments.

45. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Barry J. O'Brien whose telephone number is (703) 305-5864.

The examiner can normally be reached on Mon.-Fri. 6:30am-4:00pm.

46. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Eddie Chan can be reached on (703) 305-9712. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

47. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Aimee J. Li
Examiner
Art Unit 2183

AJL
27 November 2005

EDDIE CHAN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100
